

CLAIMS

Now, therefore, the following is claimed:

1 1. A computer system capable of performing backward error recovery,
2 comprising:
3 a memory unit having a plurality of memory locations; and
4 a memory controller configured to maintain a checksum in one of said memory
5 locations, said memory controller further configured to receive a plurality of requests
6 to update said checksum with a plurality of data values, said memory controller
7 configured to combine said checksum to each of said data values and to store each of
8 said data values, said memory controller further configured to retrieve a plurality of
9 said data values in response to a data error and to recover a previous state of a
10 particular memory location by combining each of said retrieved data values to said
11 checksum.

1 2. The system of claim 1, wherein said memory controller is configured to
2 combine each of said retrieved data values to said checksum by exclusively oring each
3 of said retrieved data values to said checksum.

1 3. The system of claim 1, wherein each of said data values represents an
2 exclusive or result between a first data value and a second data value that is replaced in
3 memory by said first data value.

1 4. The system of claim 1, further comprising a first stack and a second
2 stack, wherein said memory controller is configured to determine which protection
3 domains are associated with said data values, and wherein said memory controller, in
4 storing said data values, is configured to store in said first stack each of said data
5 values determined by said memory controller to be associated with a first protection
6 domain and to store in said second stack each of said data values determined by said
7 memory controller to be associated with a second protection domain.

1 5. The system of claim 4, wherein said memory controller is configured to
2 identify one of said protection domains in response to said data error and to select one
3 of said stacks based on which of said protection domains is identified by said memory
4 controller, and wherein each of said retrieved data values is retrieved from said
5 selected stack.

1 6. The system of claim 5, wherein said memory controller is configured to
2 correlate each of said data values with a memory controller identifier identifying
3 another memory controller that transmitted the correlated data value.

1 7. The system of claim 6, wherein each of said retrieved data values is
2 correlated with a memory controller identifier identifying a failed memory controller.

1 8. The system of claim 1, wherein said memory controller is configured to
2 identify a particular memory location in response to said data error and to identify
3 which of said data values to retrieve in response to said data error based on which of
4 said data values are associated with said particular memory location.

1 9. The system of claim 8, wherein each of said retrieved data values
2 represents an exclusive or result between a first data value and a second data value that
3 is replaced in said particular memory location by said first data value.

HP Docket No. 10001395
Page 52 of 100
Case No. 1:11-cv-00001-HP
Date Filed: 08/01/13
Page 52 of 100
Case No. 1:11-cv-00001-HP
Date Filed: 08/01/13

1 10. A computer system capable of performing backward error recovery,
2 comprising:
3 a plurality of memory units, each of said memory units having a plurality of
4 memory locations; and
5 a plurality of memory controllers configured to store to and retrieve from said
6 memory units, one of said memory controllers configured to maintain, in a checksum
7 memory location, a checksum of a checksum set, the other of said memory controllers
8 configured to receive a plurality of write requests for writing to said checksum set, said
9 other memory controllers, for each of said write requests, configured to store a first
10 data value in one of said memory locations, to retrieve a second data value from said
11 one memory location, to combine said first data value with said second data value
12 thereby forming a combined value, and to transmit said combined value to said one
13 memory controller,
14 wherein said one memory controller is configured to update said checksum
15 memory location with a plurality of combined values formed by said other memory
16 controllers and to store each of said plurality of combined values, said one memory
17 controller further configured to retrieve said stored plurality of combined values in
18 response to a data error and to combine said retrieved combined values with said
19 checksum.

1 11. The system of claim 10, wherein said memory locations within said
2 memory units are assigned to different protection domains, and wherein said one
3 memory controller is configured to determine, in a plurality of determinations, which of
4 said protection domains is associated with each of said combined values, said memory
5 controller further configured to store said plurality of combined values associated with
6 different protection domains in different stacks based on said determinations.

1 12. The system of claim 10, wherein said one memory controller is
2 configured to correlate each said plurality of combined values with an identifier that
3 identifies the memory controller that transmitted the correlated combined value to said
4 one memory controller.

1 13. The system of claim 12, wherein each of said values retrieved by said
2 one memory controller is correlated with the same identifier, said same identifier
3 identifying a failed memory controller.

1 14. The system of claim 10, wherein said memory locations within said
2 memory units are assigned to different protection domains, and wherein said one
3 memory controller is further configured to identify which of said protection domains is
4 associated with said data error and to retrieve said plurality of retrieved combined
5 values based on said identified protection domain.

1 15. The system of claim 14, wherein each of said retrieved combined values
2 is associated with said identified protection domain.

1 16. A method for performing backward error recovery, comprising the
2 steps of:

3 storing a plurality of data values within a checksum set to a plurality of memory
4 locations, said checksum set including a checksum value and a plurality of non-
5 checksum values;

6 updating one of said memory locations with a first value;

7 combining said first value to a second value to form a combined value, said
8 second value stored in said one memory location prior to said updating step;

9 updating said checksum value with said combined value;

10 storing said combined value;

11 retrieving said combined value in response to a data error;

12 combining said value retrieved in said retrieving step to said checksum value;

13 and

14 recovering a previous state of said one memory location based on said

15 combining said retrieved value step.

1 17. The method of claim 16, wherein said combining said first value step

2 includes the step of exclusively oring said first value with said second value, and

3 wherein said combining said retrieved value step includes the step of exclusively oring

4 said checksum value to said retrieved value.

1 18. The method of claim 16, further comprising the step of associating said

2 data error with a particular protection domain, wherein said retrieving step is based on

3 said associating step.

1 19. The method of claim 16, further comprising the step of assigning to a
2 respective protection domain each of said memory locations that is storing one of said
3 non-checksum values, wherein said storing said combined values step is based on
4 which protection domain is assigned to said one memory location.

1 20. The method of claim 19, further comprising the step of selecting a stack
2 based on which protection domain is assigned to said one memory location, wherein
3 said storing said combined value step includes the step of storing said combined value
4 to said selected stack based on said selecting step.

1 21. A method for performing backward error recovery, comprising the
2 steps of:
3 storing a plurality of data values within a checksum set to a plurality of memory
4 locations, said checksum set including a checksum value and a plurality of non-
5 checksum values;
6 assigning said memory locations to different protection domains;
7 storing new non-checksum values to said memory locations;
8 for each of said new non-checksum values stored to one of said memory
9 locations, combining said new non-checksum value with a value previously stored in
10 said one memory location thereby forming a combined value;
11 updating said checksum value with each combined value formed via said
12 combining step;
13 storing into memory each combined value formed via said combining step;
14 detecting a data error;
15 identifying a protection domain associated with said data error;
16 selecting a plurality of combined values formed in said combining step based on
17 said identifying step;
18 combining each of said selected combined values with said checksum value in
19 response to said data error; and
20 recovering a previous state of one of said memory locations based on said
21 combining each of said selected combined values step.

1 22. The method of claim 21, wherein said combining steps are exclusive
2 oring steps.

1 23. The method of claim 21, further comprising the step of correlating each
2 said value stored in said storing step with a respective one of said protection domains,
3 wherein each value selected in said selecting step is correlated, via said correlating
4 step, with said identified protection domain.

1 24. The method of claim 21, further comprising the step of respectively
2 correlating each said value stored in said storing step with one of said protection
3 domains, wherein said storing step is based on said correlating step.

1 25. The method of claim 24, wherein said storing step comprises the step of
2 storing into the same stack each value correlated, via said correlating step, with the
3 same protection domain.